Nos. 16-1543, 16-1545

IN THE UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

GOOGLE INC.,

Appellant

v.

INTELLECTUAL VENTURES II LLC,

Cross-Appellant

On Appeal from the Patent Trial and Appeal Board of the United States Patent and Trademark Office No. IPR2014-00787

CORRECTED OPENING BRIEF OF GOOGLE INC.

Adam M. Conrad
KING & SPALDING LLP
100 N. Tryon St., Suite 3900
Charlotte, NC 28202
Telephone: (704) 503-2600
Facsimile: (704) 503-2622
aconrad@kslaw.com

Daryl L. Joseffer
KING & SPALDING LLP
1700 Pennsylvania Ave, NW
Washington, DC 20006
Telephone: (202) 737-0500
Facsimile: (202) 626-3737
djoseffer@kslaw.com

Counsel for Google Inc.

June 3, 2016

CERTIFICATE OF INTEREST

Counsel for Google Inc. certifies the following:

1. The full name of every party represented by me is Google Inc.

2. The name of the real party in interest is Google Inc.

3. Alphabet Inc., a publicly traded company (NASDAQ: GOOG, GOOGL), has more than 10% ownership of Google Inc. No publicly held company owns 10% or more of Alphabet Inc.'s stock.

4. The names of all firms and the partners or associates that appeared for the parties now represented by me in the trial court or that are expected to appear in this Court are:

   King & Spalding LLP: Daryl L. Joseffer; Adam M. Conrad; Joshua N. Mitchell

   Kilpatrick Townsend & Stockton LLP: John C. Alemanni; Theodore G. Brown, III; Steven D. Moore

June 3, 2016

/s/ Daryl L. Joseffer
Daryl L. Joseffer

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

Rules

STATEMENT OF RELATED CASES

Pursuant to Federal Circuit Rule 47.5, Petitioner-Appellant Google states that:

(a) No appellate court has heard an appeal from the proceeding in this case; and

(b) the patentability, validity, or infringement of U.S. Patent No. 6,121,960 is or was at issue in the following cases:

- *Intellectual Ventures I LLC v. Nikon Corp.*, No. 1:11-cv-01025-SLR (D. Del.);
- *Intellectual Ventures I, LLC v. Canon, Inc.*, No. 1:13-cv-00473-SLR (D. Del.); and
- *Intellectual Ventures I LLC v. Motorola Mobility LLC*, No. 0:13-cv-61358-RSR (S.D. Fla.).

JURISDICTION

Google Inc. appeals the final written decision of the Patent Trial and Appeal Board in *inter partes* review No. IPR2014-00787. *See* 35 U.S.C. §§ 141(c) & 319. Google timely filed its petition for *inter partes* review on May 20, 2014, less than one year after Intellectual Ventures filed a civil patent-infringement complaint against Google subsidiary Motorola Mobility LLC on June 19, 2013. *See* 35 U.S.C. § 315(b); Appx36; Appx100; Appx524. The Board entered its final written decision on November 20, 2015. Appx1. Google timely filed its notice of appeal on December 22, 2015. Appx2410–12; *see also* 35 U.S.C. § 142; 37 C.F.R. § 42.102. This Court has jurisdiction under 28 U.S.C. § 1295(a)(4)(A).

INTRODUCTION

This is an appeal from an *inter partes* review of U.S. Patent No. 6,121,960 ("the '960 patent"). The patent claims a software-based system and method for showing a translucent keyboard image over a background image on a touchscreen display. During prosecution, the named inventors attempted to distinguish the '960 patent from the crowded touchscreen-display field by arguing that the claimed method involved blending or merging the foreground and background images into a single image.

That technique and the methods described for achieving it had already been described in other patents. In the proceedings below, the Patent Trial and Appeal Board found that at least two prior art references blended the pixels of a background and foreground image to create a translucent overlay. For that reason, the Board concluded that claims 19–22 and 24–30 were not patentable—a finding well supported by the record.

But the Board concluded that claims 1–3, 5, 7–10, and 12–14 are patentable over the same prior art. The Board rested that decision on a

single, unlikely ground: that the prior art did not disclose the use of "logical operators" to perform the blending.

There is nothing new, much less non-obvious, about the use of logical operators. The evidence shows that logical operators—*e.g.*, AND, NOT, OR, XOR, etc.—are the basic building blocks of digital computing and computer programming. These are the fundamental operations that computers use to manipulate binary digits (or bits). And they are the basic components that computer programmers use in combination to perform more complex operations, such as arithmetic.

It is therefore no surprise that Google's prior art references expressly disclose the use of logical operators to perform blending. Although the Board disagreed, it failed to address much of the evidence. In related district court litigation, Intellectual Ventures itself had alleged that accused products infringe by using the same blending equation described by one of the prior art references.

In any event, the use of logical operators would have been obvious as a matter of law. The '960 patent combines familiar elements according to their intended purposes to yield predictable results—a classic case of obviousness under *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S.

398, 414–16 (2007). But the Board truncated its obviousness analysis by considering only whether a prior art reference disclosed the use of logical operators for blending. By failing to take into account the extensive "background knowledge" in the art as well as "the inferences and creative steps" that a skilled programmer "would employ," the Board erred as a matter of law. *Id.* at 418.

STATEMENT OF THE ISSUES

1. Whether this Court should reverse the Board's judgment that claims 1–3, 5, 7–10, and 12–14 of the '960 patent are patentable because (a) multiple prior art references disclosed the use of "logical operators" to blend pixels of foreground and background images; and (b) the use of logical operators to perform blending would, in any event, have been obvious to a person of skill in the art.

2. In the alternative, whether this Court should remand because (a) the Board failed to consider all record evidence; and (b) the Board's final written decision was arbitrary and capricious because its reasoning is inconsistent with the Board's earlier decision instituting *inter partes* review.

STATEMENT OF THE CASE

This appeal concerns Google's petition for *inter partes* review of the '960 patent. Google has appealed the Board's confirmation of claims 1–3, 5, 7–10, and 12–14. Intellectual Ventures has cross-appealed the Board's rejection of claims 19–22 and 24–30.

A. The '960 Patent

The '960 patent claims a method for displaying a semi-transparent keyboard over a background image on a touchscreen computing device. *See* Appx114 col. 1:17–19. Figure 4, for example, illustrates a standard QWERTY keyboard superimposed over a scenic background image. *See* Appx108.

At the time of filing, it was "known in the art to superimpose a keyboard over an image" on a touchscreen. Appx114 col. 1:25–27. According to the specification, one prior art method involved rapidly switching between the background and foreground images many times per second, creating the illusion of an overlapping image (akin to a child's thaumatrope toy, where images on different sides of an object appear to be combined when the object is spun or twirled rapidly). *See id.* col. 1:33–43. During prosecution, the examiner cited a second prior art method, which created a stippled image by interspersing pixels of

the foreground image among the pixels of the background image. Appx1062 col. 2:13–16.

The '960 patent's alleged advance was to merge pixels from the two images to produce a single, blended image. *See* Appx266–67; Appx115 col. 4:42–46. In other words, rather than choose some pixels from one image and some from the other, the claimed invention would allow "individual pixels of the touch-activated input device [to] be dedicated simultaneously to both" the background and the keyboard images. *Id.* One named inventor referred to the process of merging the two images as "alpha blending." Appx1485.

The claims at issue in Google's appeal are independent claim 1 and its dependent claims 2, 3, 5, 7–10, and 12–14. For present purposes, claim 1 is representative of those claims. It recites superimposing a "representation" of a "key" over a "main image" to display "a composite image." Appx119 col. 11. The claim refers to the process of forming the composite image as "variable-pixel control," which "simultaneously" displays pixels of the two images. *Id.* The claimed variable-pixel control uses "logical operators" to blend or merge

pixels of the background and foreground images. Appx116 col. 5:1–19; *see also* Appx119 col. 12:25–26. The claim reads in full:

> 1. A screen peripheral system, comprising:
>
> a computing device for providing a main image; and
>
> a touch-activated input device for generating and displaying a composite image visible to a user of the screen peripheral system, the touch-activated input device comprising a plurality of pixels, the composite image simultaneously including:
>
> a representation of at least one key, the representation of at least one key activating an input function; and
>
> the main image provided by the computing device, the representation of at least one key being laid over the main image;
>
> wherein the screen peripheral system implements variable-pixel control to form the representation of at least one key and to form the main image, the variable-pixel control causing pixels selected to form the representation of at least one key in the composite image to depend on and be activated simultaneously with pixels selected to form the main image, such that the main image and the representation of at least one key are displayed simultaneously to form the composite image;
>
> further wherein the variable-pixel control includes *logical operators to provide different blending/merging effects* such that individual pixels of the touch-activated input device can be dedicated simultaneously to both the main image and the representation of at least one key.

Appx119 col. 12:2–29 (emphasis added).

Logical operators, as referenced in this claim, represent the basic operations that computers use to manipulate values at the bit level. *See*

Appx9. (In a computer, a bit is the smallest unit of data — a binary one or zero. In a digital electronic computer, a bit is modeled as an electrical current that is either on (1) or off (0). *See* Appx1572.) Logical operators take input values and use them to produce output values based on predefined rules. Appx1257. Examples include the Boolean logical operators AND, OR, NOT, and XOR. *See* Appx9, 1257.

For example, NOT takes a single input bit and outputs its opposite, so NOT 1 → 0, and NOT 0 → 1. The operator AND takes two input bits, and if *both* of them are 1s, the output is 1; otherwise it is 0: 1 AND 1 → 1, but 1 AND 0 → 0. *See id.* The operator OR takes two input bits, and if *either* of them is a 1, it outputs a 1: 1 OR 0 → 1, 1 OR 1 → 1, but 0 OR 0 → 0. *See* Appx1216–17. The operator XOR, which stands for "exclusive OR," is like OR except that it outputs a 1 only if exactly one (not both) of the bits is 1: 1 XOR 0 → 1, but 1 XOR 1 → 0. Computer processors use these logical operators, in combination, to perform math. *See* Appx1302–03; Appx1569.

The specification emphasizes the breadth and flexibility that "logical operations" can have in creating "different blending or merging effect[s]." Appx116 col. 5:26–27. For example, some logical operations

"just copy the source in to the destination or just fill it with zeros or ones"; other logical "operations can be used to create a number of different effects" for merging the two images. *Id.* col. 5:27–33. Logical operators can also "be combined as needed to produce an even greater variety of visual effects." *Id.* col. 5:34–37.

The specification includes a table of "typical" logical operators and their effects:

		Boolean	
Source (S)	1 1 0 0 1 1 0 0		
Destination (D)	1 0 1 0 1 0 1 0		
Mask (M)	1 1 1 1 0 0 0 0	Operation	Operation
Result	0 0 0 0 0 0 0 0	0	Blackness
	0 0 0 1 0 0 0 1	~(S|D)	Not source erase
	0 0 1 1 0 0 1 1	~S	Not source copy
	0 1 0 0 0 1 0 0	S & ~D	Source erase
	0 1 0 1 0 1 0 1	~D	Destination invert
	0 1 0 1 1 0 1 0	M ^ D	Mask invert
	0 1 1 0 0 1 1 0	S ^ D	Source invert
	1 0 0 0 1 0 0 0	S & D	Source and
	1 0 1 1 1 0 1 1	~S|D	Merge paint
	1 1 0 0 0 0 0 0	M & S	Merge copy
	1 1 0 0 1 1 0 0	S	Source copy
	1 1 1 0 1 1 1 0	S|D	Source paint
	1 1 1 1 0 0 0 0	M	Mask copy
	1 1 1 1 1 0 1 1	M|~S|D	Mask paint
	1 1 1 1 1 1 1 1	1	Whiteness

Appx115 col. 4:66, Appx116 col. 5:1–19. The table shows three 8-bit bit fields, which are labeled Source, Destination, and Mask. (A "bit field" is a string of logical bits (binary 1s and 0s) that allows operations both on single bits in the field and on the field as a whole.) The table lists

fifteen outcomes from the application of logical operators to one or more of these three bit fields. *See id.*

The Mask bit field allows multiple bits in the Source or Destination bit field to be controlled using a single bitwise operation. For example, one common use for a mask bit field in computer graphics is to indicate which portions of a foreground image are transparent. *See* Appx1796 (defining mask as "a binary value used to selectively screen out or let through certain bits in a data value"). Creating a transparency mask in the shape of the foreground image and then combining it with the background image using the logical operator AND creates a "cutout" in the background image. *See id.* Then, the foreground and background can be combined using the logical operator OR. *See id.*

The '960 patent's "Source Invert" operation ("S ^ D") is an XOR ("exclusive OR") combination of the Source and Destination bit fields. (Programming languages use the caret symbol to denote a bitwise XOR operator.) The XOR operator produces a value of 1 for each bit where the Source and Destination bit fields contain different values, and a value of 0 wherever Source and Destination are the same. Appx116 col.

5:12. The table also discloses logical operations called "Source copy" and "Mask copy," in which the output bit field is simply a copy of the Source and Mask bit fields, respectively. *Id.* col. 5:15 & 17.

B. The Prior Art

Google's petition for *inter partes* review identified three prior art references that blended or merged pixels to produce a composite image.

1. The Gough Reference (U.S. Patent No. 5,638,501)

Gough discloses a "method and apparatus for providing a translucent overlay image over a base image on the screen of a computer system," such as a screen that "can receive input from a pen or stylus." Appx348 (Abstract). Several of the figures in Gough show semitransparent foreground keyboards overlaid on a background graphical user interface. *Compare, e.g.*, Appx352 fig. 4, *with* Appx108 fig. 4.

According to Gough's specification, "[t]he step of displaying an overlay image preferably involves the blending of an overlay image with the base image." Appx371 col. 2:48–49. Among other methods, Gough discloses combining foreground and background image bitmaps using a Boolean logical "OR" operation. *See* Appx361 figs. 10a–10f. Figures 10a and 10b show two four-by-four pixel bitmaps:

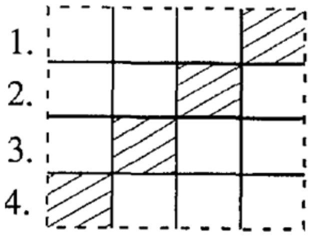


Fig. 10a

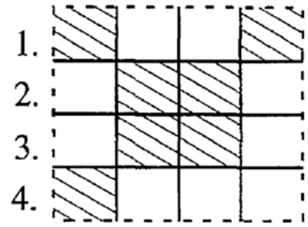


Fig. 10b

Id. Each bitmap is divided into four one-by-four rows. *Id.* Corresponding rows from each of the two bitmaps are then combined (in Figures 10c, 10d, 10e, and 10f):

Fig. 10c

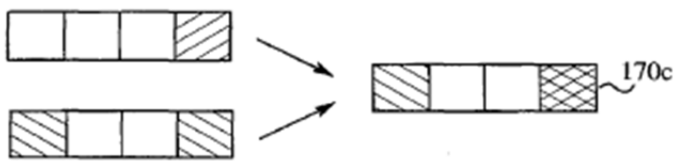


Fig. 10d

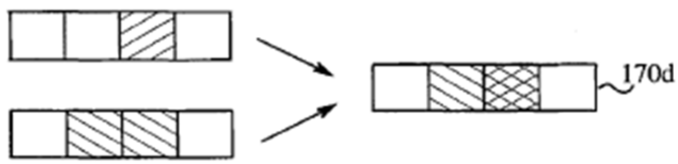


Fig. 10e

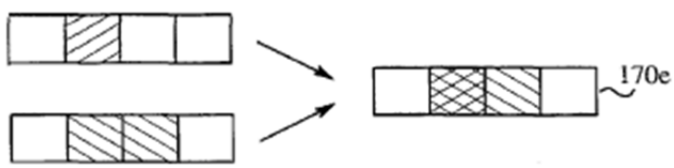
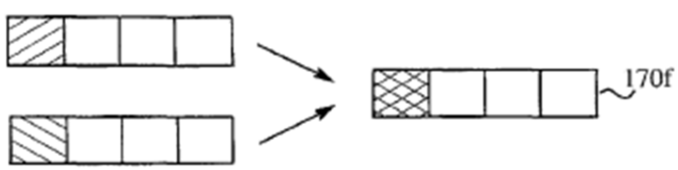


Fig. 10f



Id. These figures show a logical OR operation: where either input bitmap has an "on" pixel, the output bitmap also has an "on" pixel; and

where both input bitmaps have an "off" pixel, the output bitmap has an "off" pixel. *See id.*

Gough also discloses two additional methods of merging an overlay and background image. First, it uses a "color look-up table," which "detail[s] each possible combination of bits from" two given pixels "and what the resultant blended value is." Appx377 col. 14:13–22. In other words, the table stores a value for every outcome of a given logical operator for two bitmap values. *See id.* col. 14:13–22. Second, Gough describes a blending engine that makes identical copies of bitmaps from two separate image buffers. *See* Appx362 fig. 11 & Appx363 fig. 12.

2. The Buxton Reference (U.S. Patent No. 6,118,427)

Buxton discloses using "variable transparency to merge images (or layers) of objects onto a graphical display." Appx422 (Abstract). One exemplary figure shows a translucent three-by-four keyboard superimposed over an image of a truck. Appx424 fig. 1.

Buxton discloses two methods of merging images: continuous and discrete algorithms. *E.g.*, Appx439 col. 16:7–10. "Continuous algorithms" include "alpha blending" and "filtered transparency."

Appx440 col. 17:3–4. These "flexible" techniques allow for "any transparency level." *Id.* col. 17:7–9.

Buxton describes "an alpha blending algorithm" as its preferred embodiment:

$$I = \alpha I_1 + (1 - \alpha) I_2.$$

Id. col. 17:17–33. This equation combines color values from a "foreground image pixel" and a "background image pixel." *Id.* col. 17:25–26. The variable I stands for the resulting color value; I_1 is the foreground image pixel color value, and I_2 is the background image pixel color value. *See id.* The "α [alpha] value[]" indicates the degree of transparency of the foreground image pixel ("0=clear and 1=opaque"). *Id.* col. 17:19, 26–28. In the equation, the color values of the foreground and background pixels are set in a simple inverse relationship: where the foreground pixel is fully opaque, the background pixel is fully transparent. *See id.* Similarly, a 75% opaqueness value for the foreground pixel will combine with a 25% opaqueness value for the background pixel. *See id.*

Buxton also discloses the use of "[d]iscrete algorithms." Appx439 col. 16:11. These algorithms include "dithering, stippling, XORing, and

'screen-door transparency.'" *Id.* col. 16:15–17. Discrete algorithms "are generally simple to specify" and "can be predefined and saved." *Id.* col. 16:37–38.

3. The Martin Reference (U.S. Patent No. 5,148,155)

Martin discloses a "computer system having a digitizing tablet overlaying the display screen." Appx379 (Abstract). The digitizing tablet is used to allow input on screen-simulated input devices, including keyboards, on the computer screen using a "pointer-type device." Appx394 col. 3:17–49; *see also id.* col. 4:13–14 & Appx391 fig. 10.

Although the preferred embodiment in Martin is an opaque overlay, the patent also discloses blending foreground and background images on "a pixel-by-pixel basis." Appx397 col. 9:6. To do so, Martin uses "a set of image-combining arrangements" comprising Boolean logical operators "such as AND, OR, XOR, NOR, and NAND." *Id.* col. 9:5–13.

C. Procedural History

On June 19, 2013, Intellectual Ventures filed a complaint in the Southern District of Florida alleging that mobile phone handsets manufactured, imported, and sold by Google subsidiary Motorola

Mobility infringed seven patents, including the ’960 patent. Appx524–41. In its infringement contentions, Intellectual Ventures alleged that the accused handsets use an alpha blending algorithm to merge foreground and background images. *See* Appx1370 (alleging “the blending is the combination of the [alpha value] of the destination plus [1 – the alpha value] of the source” (brackets in original)).

1. Institution Decision

Google petitioned for *inter partes* review of claims 1–3, 5, 7–10, 12–14, 19–22, and 24–30 of the ’960 patent. On November 24, 2014, the Board issued a decision instituting review. Appx869–91. The Board concluded that Google was reasonably likely to prevail on two grounds: first, that Gough anticipated the challenged claims; and second, that Buxton (in combination with two other references)[1] rendered the claims obvious. Appx889.

The Board declined to institute review as to Martin. The Board did not determine that Google had failed to demonstrate a reasonable

[1] U.S. Patent No. 5,617,114 (Bier) teaches, among other things, the interchangeability of digitizer tablets, mouse input, and touchscreens. Appx920 col. 7:41–45. U.S. Patent No. 6,317,128 (Harrison) discloses additional variable-transparency methods for combining foreground and background images. *See generally* Appx484–501. Neither is directly relevant to the issues on appeal.

probability that it would prevail on that reference. Instead, the Board denied review on the ground that Martin was "redundant." *Id.* (citing *Liberty Mut. Ins. Co. v. Progressive Cas. Ins. Co.*, No. CBM2012-0003 (PTAB Oct. 25, 2012) (Paper 7)).

2. Final Written Decision

In its final written decision, the Board held that independent claims 19 and 26, as well as their dependent claims, were anticipated by Gough and also (with the exception of claim 21) rendered obvious by Buxton. *See* Appx28–29. Claim 19 claims a "method of superimposing a representation of at least one key over a main image provided by a computing device." Claim 26 is generally a means-plus-function version of claim 1. Neither of those claims contains a "logical operators" limitation. *See* Appx120.

Among other things, the Board concluded that "Gough discloses variable-pixel control" and that "Gough's blending process involves blending data from blendable units representing a base or main image with data from blendable units representing an overlay keyboard image, so that the base image can be seen through the translucent keyboard image." Appx19–20. The Board further noted that

Intellectual Ventures had conceded that Buxton "generates a resulting blended image in which pixels have contributions from *both* the foreground image pixels and background image pixels." Appx26 (quoting Appx1038–39) (emphasis in original).

The Board rejected Intellectual Ventures' contention that claims 19 and 26 required that a composite image contain both unblended pixels from the original images *and* blended pixels that were combinations of both foreground and background. *See* Appx16–18. The Board concluded that the claims contain no such requirement, Appx18–19, and that in any event the prior art disclosed composite images with both unblended and blended pixels. Appx19 (discussing Gough); *see also* Appx25–26 (discussing Buxton). The Board thus concluded that claims 19 and 26 and their dependent claims were anticipated and obvious.

The Board reached a different conclusion as to independent claim 1 and its dependent claims 2, 3, 5, 7–10, and 12–14. *See* Appx28–29. The Board held those claims patentable on a single ground: that Gough and Buxton do not disclose the use of logical operators to blend the

foreground and background pixels. *See* Appx15–16, 24; *see also* Appx119 col. 12:25–26.

The Board construed "logical operators" to mean "operator[s] that manipulate[] binary values at the bit level." Appx9. According to the Board, that construction is consistent with the table in the specification "summarizing typical operations that can be used to provide blending/merging effects." *Id.*; *see also* Appx116 col. 5:1–19. The Board declined to limit "logical operators" to "Boolean logic operators," which Intellectual Ventures had proposed, noting that Boolean operators are a subset of logical operators. *See* Appx8–9.

The Board appeared to agree that "Gough *could* be envisioned as using logical operators." Appx16 (emphasis in original). But the Board found no "express disclosure" of logical operators in "Gough's description of the blending process depicted in Figures 10a–10f" or in its "description of using the color look-up table." *Id.*

With respect to obviousness in light of Buxton and other references, the Board determined that Buxton's alpha blending equation "involves arithmetic operations" and that such operators "differ from logical operations." Appx24. The Board did not explain

how the two differ in any relevant respect, nor did it explain why arithmetic operations do not satisfy its construction of the claim term "logical operators." *Id.* Although Intellectual Ventures had argued before the district court that the same alpha blending equation infringed the claims, the Board held that Intellectual Ventures was not estopped from taking the opposite position for purposes of patentability. *See* Appx25.

The Board ended its obviousness analysis there. Having found that arithmetic and logical operators differed, the Board held the claims to be non-obvious without considering whether it would have been obvious to one of skill in the art to bridge the alleged gap. *See* Appx24–25. Nor did the Board address whether Buxton's alternative use of XORing (in its discrete algorithm embodiment) satisfied the logical operators limitation. *See* Appx22.

The Board offered no explanation for implicitly departing from its earlier finding that Martin, which explicitly disclosed the use of logical operators, was redundant with Gough and Buxton, which the Board found not to contain that disclosure.

SUMMARY OF THE ARGUMENT

The use of logical operators to blend pixels was disclosed in the prior art and obvious as a matter of law.

I. To a person skilled in the art, Gough's figures 10a through 10f expressly depict the output of a Boolean OR operator, as Google's expert explained. Although the Board concluded that Gough's figures and accompanying description do not "expressly disclose[]" the use of logical operators, Appx16, the Board cited no evidence to support that conclusion or to discredit Google's expert testimony.

Rather, the Board evidently turned the anticipation inquiry into a word search for "logical operators" in the prior art. That was error because an anticipatory reference need not use exactly the same words as the challenged claim. *See, e.g.*, *WhitServe, LLC v. Computer Packages, Inc.*, 694 F.3d 10, 21 (Fed. Cir. 2012).

Two other embodiments in Gough also disclose logical operators under correct legal principles. First, Gough describes a color look-up table of each possible output of a bitwise operation, which by definition involves "manipulat[ing] binary values at the bit level" (the Board's definition of logical operators, *see* Appx9). Second, Gough describes

block copy operations that function exactly like bitwise copy operations disclosed in the ’960 patent.

II. The claims at issue were also obvious in light of Buxton and other references.

A. The Board found that Buxton’s alpha blending algorithm disclosed arithmetic operators, and that arithmetic operators are different from logical operators. Google had shown, however, that Buxton’s arithmetic operation was a combination of logical operators. Indeed, computers perform arithmetic operations by using one or more logical operators. The Board did not address that evidence, which should have been dispositive.

Intellectual Ventures itself argued in the district court that the same alpha blending equation would infringe the ’960 patent’s claims. *See* Appx1370. The Board erred by allowing Intellectual Ventures to take the opposite position for purposes of patentability.

In any event, Buxton’s discrete algorithms embodiment separately discloses the use of logical operators, stating expressly that “XORing” may be used to blend images. It is undisputed that XOR is a Boolean logical operator. The Board did not address this disclosure.

B. Finally, the Board failed to conduct a full obviousness analysis. Logical operators are the ubiquitous building blocks of computer science. Indeed, one of the '960 patent's inventors conceded that the patent simply uses "what was available" in commonly available Windows tools for computer programmers. Appx1488. Even if Buxton had not disclosed the use of logical operators for blending, it would have been obvious as a matter of law for a skilled artisan to use those commonly available tools for their familiar and intended purposes.

III. In any event, the Board's final written decision was arbitrary and capricious because it departed without explanation from its earlier finding that the instituted grounds were redundant of the Martin ground. If Gough and Buxton do not disclose logical operators, as the Board concluded, then Martin is not redundant, because it unquestionably discloses logical operators.

STANDARD OF REVIEW

Under the Administrative Procedure Act, this Court sets aside "actions of the Board that are arbitrary, capricious, an abuse of discretion, or otherwise not in accordance with law." *Microsoft Corp. v. Proxyconn, Inc.*, 789 F.3d 1292, 1306 (Fed. Cir. 2015) (internal

quotation omitted). Under those standards, the Board's conclusions of law are reviewed *de novo*, and its findings of fact are reviewed for substantial evidence. *Id.* at 1297.

Anticipation is a question of fact and thus reviewed for substantial evidence. *Synopsys, Inc. v. Mentor Graphics Corp.*, 814 F.3d 1309, 1317 (Fed. Cir. 2016). The ultimate conclusion of obviousness is a question of law reviewed *de novo*. *Agrizap, Inc. v. Woodstream Corp.*, 520 F.3d 1337, 1343 (Fed. Cir. 2008).

The Board's entitlement to deferential review of factual questions brings with it a correlative responsibility to consider the evidence and explain its reasoning. Under the arbitrary-and-capricious standard, "the Board's opinion must explicate its factual conclusions, enabling [the Court] to verify readily whether those conclusions are indeed supported by 'substantial evidence' contained within the record." *In re Gartside*, 203 F.3d 1305, 1314 (Fed. Cir. 2000) (citation omitted). Meaningful judicial review requires the agency to "make the necessary findings and to provide an administrative record showing the evidence on which the findings are based, accompanied by the agency's reasoning

in reaching its conclusions." *In re Sang-Su Lee*, 277 F.3d 1338, 1342 (Fed. Cir. 2002).

ARGUMENT

The Board agreed that both Gough and Buxton disclosed "variable-pixel control" and blending pixels of foreground and background images prior to the '960 patent. *See* Appx20, Appx26. Yet the Board confirmed patentability of some claims on the surprising ground that the prior art did not use "logical operators" to perform the blending. Logical operators are basic, well-known elements of computer science—hardly a hallmark of novelty, much less non-obviousness.

I. Gough Anticipated The Challenged Claims.

The Board recognized that "Gough's blending process involves blending data from blendable units representing a base or main image with data from blendable units representing an overlay keyboard image, so that the base image can be seen through the translucent keyboard image." Appx19. Google further demonstrated that each of Gough's three embodiments for that blending process uses logical operators. *See* Appx58–59; Appx1175–77.[2]

[2] Although the Board stated that "the detailed analysis set forth in the Reply" concerning Gough's disclosure of logical operators was not

Figures 10a through 10f in Gough expressly disclose the use of a logical OR operator to blend the pixels of two images. Gough's specification explains that the blending engine arranges each unit into a "four-by-four matrix" or "array." App361 figs. 10a–10f; Appx375 col. 10:23–28. The engine then blends corresponding rows from each array "to form a blended row." Appx375 col. 10:33. A pixel in the blended row is "off" when the corresponding pixel is "off" in both input rows, and a pixel in the blended row is "on" if the corresponding pixel is "on" in either input row. *See* App361; *see also* Appx564–65. As Google's expert explained, that is the outcome of "an OR operation," which is a standard Boolean logical operator. Appx1216–17.

Indeed, the '960 patent describes this very same OR operation with binary numbers, where "off" and "on" are represented with binary 0 and 1 respectively. *Compare* Appx361 figs. 10a–10f (Gough), *with*

---

also "made in the Petition," the Board did not decline to consider Google's showing for that reason. *See* Appx15. Instead, it addressed Google's argument on its merits. *See* Appx15–16. And rightly so, as there was no waiver. Google's petition relied on Gough's disclosures as satisfying the "logical operators" limitation. *See* Appx58–59. When Intellectual Ventures relied heavily on that limitation in its Answer, Google responded in its Reply with the more "detailed showing" regarding the same Gough disclosures. *Compare id. with* Appx1175–77. That is what replies are for.

Appx116 col. 5:16 ('960 patent); *see* Appx639–40. Gough and the '960 patent thereby disclose the use of the same logical operator for the same purpose, removing any doubt on anticipation. *See In re Gleave*, 560 F.3d 1331, 1334 (Fed. Cir. 2009) (anticipation shown when elements are "combined in the same way as in the claim").

The Board's contrary conclusion is wrong as a matter of law. The Board cited no evidence to support it and provided no rebuttal to Google's showing. *See generally* Appx15–16.

There is no such evidence. Intellectual Ventures relied on the conclusory declaration of its expert, Dr. Rosenberg. *See* Appx1021. After conceding that "Gough describes blending together pixels of the base image with pixels of the overlay image," Dr. Rosenberg opined "that Gough does not describe logical operators for performing the blending." Appx1124. Dr. Rosenberg provided no basis for that conclusion; he simply repeated himself, stating that "Gough does not describe how the blending is performed and does not describe using logical operators or Boolean logic." *Id.* Such conclusory expert testimony cannot support the Board's finding. *See ActiveVideo Networks, Inc. v. Verizon Commc'ns, Inc.*, 694 F.3d 1312, 1327 (Fed. Cir.

2012) (conclusory expert testimony on obviousness is "not sufficient"); *TechSearch, L.L.C. v. Intel Corp.*, 286 F.3d 1360, 1372 (Fed. Cir. 2002) (conclusory expert statements are insufficient to raise a genuine issue of material fact); *see also PharmaStem Therapeutics, Inc. v. ViaCell, Inc.*, 491 F.3d 1342, 1361 (Fed. Cir. 2007) (rejecting testimony contrary to express disclosure of prior art reference).

The Board stated that, "[a]t best, Petitioner's arguments suggest how Gough *could* be envisioned as using logical operators." Appx16 (emphasis in original). But the Board concluded there was no "express" disclosure in Gough. *Id.* The Board thereby appears to have found "Gough's description" to be insufficiently "express" because it did not use the words "logical operators." But anticipation does not require magic words. *See WhitServe, LLC v. Computer Packages, Inc.*, 694 F.3d 10, 21 (Fed. Cir. 2012); *Gleave*, 560 F.3d at 1334.

Indeed, a prior art reference's drawings or figures may suffice to demonstrate anticipation. *See In re Watts*, 58 F.2d 841, 842 (C.C.P.A. 1932) ("[D]escription for the purposes of anticipation can be by drawings alone as well as by words."); *see also In re Aslanian*, 590 F.2d 911, 914 (C.C.P.A. 1979) ("[A] drawing in a utility patent can be cited against the

claims of a utility patent application even though the feature shown in the drawing was . . . unexplained in the specification of the reference patent."). And what a person of ordinary skill would understand from the prior art may be proved by expert testimony, as Google did here. *E.g.*, *Verizon Servs. Corp. v. Cox Fibernet Va., Inc.*, 602 F.3d 1325, 1339 (Fed. Cir. 2010) (expert "detailed how the prior art disclosed each of the claim elements"); *see also* Fed. R. Evid. 702(a).

Moreover, Google demonstrated that Gough's alternate embodiments also use logical operators to perform the blending process. For example, Gough describes a "color look-up table" that lists "each possible combination of bits from the 4 bit screen and the 4 bit overlay." Appx377 col. 14:13–22. Google's expert explained that this "is a form of a logical truth table." Appx1215. Implementing logical operators with truth tables is "well-known in the art," *id.*, a point IV's expert conceded. He described such truth tables as characteristic of logical operators and elaborated that in an "AND" truth table, for example, "[i]f you have one and a one, the truth table says the answer is one." Appx1257–58.

The Board again concluded, without explanation, that Gough's look-up table disclosure was insufficiently express. *See* Appx16. But

the undisputed evidence shows that the described look-up table displays every possible output of a bitwise operation using two four-bit bit fields. *See* Appx570. By definition, that involves "manipulat[ing] binary values at the bit level," which is the Board's definition of a logical operator. Appx9. Significantly, the Board never explained how look-up tables evade that claim construction. Under the correct legal principles discussed above, it is irrelevant that Gough does not use the words "logical operator" in addition to describing it through figures.

Gough also discloses a blending engine that copies sets of bits from two separate image buffers into the engine. *See, e.g.*, Appx357; Appx375 cols. 9:66–10:11. Google's expert explained that these block-move operations are the same as the Source Copy and Mask Copy operations described in the '960 patent. Appx1216; *compare* Appx362 fig. 11 & Appx363 fig. 12, *with* Appx116 col. 5:15 & 17. As a matter of law, this shows anticipation because it is exactly what the patent describes as a claimed embodiment. *See In re Lukach*, 442 F.2d 967, 970 (C.C.P.A. 1971) ("[D]escription of a single embodiment of broadly claimed subject matter constitutes a description of the invention for anticipation purposes."). The Board did not address this argument.

Because Gough disclosed the use of logical operators, the Board's judgment must be reversed. Alternatively, this Court should vacate and remand for a reasoned explanation. As noted above, the Board evidently relied on an incorrect legal standard by looking for the words "logical operators," and it did not analyze the evidence showing that a person skilled in the art would understand that all three embodiments used logical operators.

An agency must assess "both the evidence that justifies and [the evidence that] detracts from [the] agency's opinion." *Princeton Vanguard, LLC v. Frito-Lay N. Am., Inc.*, 786 F.3d 960, 970 (Fed. Cir. 2015). Moreover, "the Board must not only assure that the requisite findings are made, based on evidence of record, but must also explain the reasoning by which the findings are deemed to support the agency's conclusion." *Sang-Su Lee*, 277 F.3d at 1344. The Board's failure to offer any reasoned explanation on the crucial points noted above is arbitrary and capricious. *See id.*; *see also Cutsforth, Inc. v. MotivePower, Inc.*, No. 2015-1316, 2016 WL 279984, at *3 (Fed. Cir. Jan. 22, 2016) (citing *Sang-Su Lee*).

II. The Buxton Combinations Render The Challenged Claims Obvious.

The claims were also obvious over Buxton in combination with Bier or Harrison. The Board disagreed solely based on its view that Buxton did not expressly disclose the use of logical operators for blending two images. Buxton did. But even if it had not, any difference between Buxton's alpha blending algorithm and logical operators would not be a patentable distinction. The use of logical operators in computer programming was, to put it mildly, well known.

A. Buxton Discloses Blending Using Logical Operators.

Intellectual Ventures conceded that Buxton "generates a resulting blended image in which pixels have contributions from *both* the foreground image pixels and background image pixels." Appx26 (quoting Appx1038–39) (emphasis in original). Buxton teaches performing the blending process in two ways: continuous algorithms including an alpha-blending algorithm; and discrete algorithms. Appx439 col. 16:9–10. Each method uses logical operators.

1. Buxton's Alpha Blending Algorithm Uses Logical Operators.

Buxton's "alpha-blending algorithm" consists of an arithmetic equation: $I = \alpha I_1 + (1\text{–}\alpha)I_2$. Appx440 col. 17:17–23. Undisputed

evidence shows that computers implement arithmetic operations such as addition, subtraction, and multiplication using one or more logical operators. Appx1221 ¶ 60; Appx1222 n.8.

Logical operators are the "building blocks" of computer processors. Appx1572; *see also* Appx1222. Computers perform arithmetic by processing binary values in components, such as half adders, full adders, multipliers, and dividers, that comprise logical operators. Appx1302–10. For example, a half adder combines Boolean logical operators AND and XOR. *See* Appx1568–69; *see also* Appx1222.

In short, Buxton's equation requires a combination of logical operators to perform the mathematical operation. If one logical operator satisfies the claim limitation, it necessarily follows that a combination of logical operators satisfies it as well. Indeed, the '960 patent confirms that "various logical operations can be combined as needed." Appx116 col. 5:34–35.

Moreover, a computer performing arithmetic operations will, by definition, "manipulate binary values at the bit level," which is how the Board defined "logical operator." Appx24; *see, e.g.*, Appx1302 (explaining how computer hardware performs addition using "bits . . .

as input" and producing "sum" and "carry bit[s]"). For example, when a computer performs the decimal arithmetic 2 + 3 = 5, it represents those values internally as the binary values 10 + 11 → 101, and it manipulates the first two values using its circuitry components to arrive at the third.

The Board's distinction between mathematical and logical operators is especially meaningless because the foundation of modern computer science is that mathematical operations can be expressed as bitwise logical operations: logical operations and arithmetic operations are, in all important respects, two methods of expressing the same underlying math. *See* Appx1302 (explaining how computers perform arithmetic using bitwise logical operations); Appx1816–17 (showing that arithmetic instructions are merely different levels of abstraction for bitwise binary operations).

Intellectual Ventures is in no position to dispute this point. In the underlying district court litigation, Intellectual Ventures alleged that the accused handsets infringe by using the same alpha blending algorithm disclosed by Buxton: "the blending is the combination of the [alpha value] of the destination plus [1 – the alpha value] of the source."

Appx1370 (brackets in original). Although Intellectual Ventures used words instead of symbols, the meaning is the same, and this on-point admission is dispositive. "'That which infringes if later anticipates if earlier.'" *Brown v. 3M*, 265 F.3d 1349, 1352 (Fed. Cir. 2001).

The Board determined that Intellectual Ventures was not judicially estopped from taking different positions for purposes of infringement and patentability. *See* Appx25. But claims cannot be construed more broadly for infringement than for patentability. *Southwall Techs., Inc. v. Cardinal IG Co.*, 54 F.3d 1570, 1578 (Fed. Cir. 1995). And accused infringers are entitled to take a patentee's infringement allegations at face value for purposes of arguing that those allegations would invalidate the patent, which is all that Google did here. *See Vanmoor v. Wal-Mart Stores, Inc.*, 201 F.3d 1363, 1366 (Fed. Cir. 2000) ("burden [of proving invalidity] was satisfied by [patentee's] allegation that the accused cartridges infringe"). There is no reason to apply a different rule here simply because the parties are litigating in two forums instead of one. Doing so would only encourage abusive tactics.

In any event, the Board did not explain its statements that "arithmetic operations . . . differ from logical operations" and that Google "has not shown persuasively whether any logical operations overlap with any arithmetic operations." Appx24. With the evidence discussed above, Google did just that. The Board did not provide any reasoned explanation for rejecting that evidence. Nor did the Board discuss the '960 patent's statement that "various logical operations can be combined as needed" to produce blending effects. Appx116 col. 5:34–35. At a minimum, this Court should remand for a "full and reasoned" explanation. *Sang-Su Lee*, 277 F.3d at 1342.

2. Buxton's Discrete Algorithms Embodiment Discloses Blending With Logical Operators.

Google also demonstrated that Buxton's discrete algorithms method uses logical operators. Appx82 (quoting Buxton); Appx690–91 (tying XORing disclosure in Buxton to the logical operators claim limitation); Appx1182. Indeed, Buxton could not be more plain: blending can be achieved through "dithering, stippling, *XORing*, and 'screen-door transparency.'" Appx439 col. 16:15–17 (emphasis added). The XOR (exclusive OR) operator is undisputedly a Boolean logical operator. *E.g.*, Appx1257:9–11 & 22–23.

The Board never addressed this explicit disclosure in Buxton—despite acknowledging that "Buxton discloses two techniques for merging or blending foreground and background images." Appx22. Buxton's clear disclosure of the XOR logical operator requires reversal. At a minimum, this Court should remand for a "full and reasoned" explanation. *Sang-Su Lee*, 277 F.3d at 1342.

B. Any Perceived Difference Between Logical And Arithmetic Operators Is Not A Patentable Distinction.

Even if the Board were otherwise correct, "the mere existence of differences between the prior art and an invention does not establish the invention's nonobviousness." *Dann v. Johnston*, 425 U.S. 219, 230 (1976). The Board failed to consider the ultimate legal question—namely, whether it would have been obvious to one skilled in the art to "bridge the differences" between the prior art and the claimed invention. *Ohio Willow Wood Co. v. Alps South, LLC*, 735 F.3d 1333, 1343 (Fed. Cir. 2013). Here, the "gap" (if any) between the patent and the prior art is small and unworthy of patent protection. *See Scanner Techs. Corp. v. ICOS Vision Sys.*, 528 F.3d 1365, 1382 (Fed. Cir. 2008).

Simply put, if the use of logical operators to perform a computing application is a mark of patentability, then section 103 is a dead letter.

Logical operators are ubiquitous in computer science: they are the basic building blocks of computer processing and the essential tools of a computer programmer. *See* Appx1221 ¶ 60; Appx1302–10; Appx1568–69; Appx1572. As one of the named inventors testified, the '960 patent simply made use of the tools that were readily "available in 1990-whenever in the [Microsoft] Windows API[]"—that is, an Application Programming Interface commonly available to anyone in the field. Appx1488. Intellectual Ventures' expert also testified that he had programmed with logical operators as early as 1978 and had used logical operators to create composite images in the 1980s. Appx1256–57. Such long-known, fundamental concepts are well within the "background knowledge possessed by a person having ordinary skill in the art." *KSR*, 550 U.S. at 418.

That makes this a textbook case of obviousness. "The combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *See id.* at 416. Neither the Board nor Intellectual Ventures identified any unexpected result from the patent's use of logical operators. No surprise there: By their very nature, *logical* operators apply predefined rules to produce

well-known and entirely predictable results. *E.g.*, pp. 7–8, *supra*; Appx1257–58 (Intellectual Ventures' expert explaining truth tables).

Moreover, there is no relevant difference between logical operators and Buxton's arithmetic equation for alpha blending. As discussed above, arithmetic operations are combinations of Boolean operations. *See* Section II.A.1, *supra*. Thus, the difference between expressing a mathematical equation using arithmetic operators as opposed to bitwise operators is at best a formal distinction. *See, e.g.*, Appx1816–17 (explaining how computer "translate[s]" addition instruction into binary, bitwise operation).

As a result, a skilled programmer would be able to use either arithmetic or logical operators to perform the same functions. Buxton itself bears this out. For certain values, Buxton's alpha-blending equation replicates operations in the '960 patent's table of exemplary logical operations. *Compare* Appx440 col. 17:17–28 (Buxton) *with* Appx116 col. 5:1–19 ('960 Patent). For example, an alpha value of 1 (*i.e.*, full opaqueness) collapses Buxton's equation to $I = I_1$, producing output identical to the '960 patent's "Source copy" operation. *See* Appx1180–81; Appx1223 ¶ 62. It is no surprise that the named

inventors of the '960 patent and Buxton both described their methods as "alpha blending." Appx1485.

At a minimum, the use of logical operators would have been obvious to try. *See KSR*, 550 U.S. at 421; *In re Kubin*, 561 F.3d 1351, 1359 (Fed. Cir. 2009). Using a computer to blend two pixels is a problem with "a finite number of identified, predictable solutions," including Boolean logical operators, arithmetic operators, and look-up tables. *See KSR*, 550 U.S. at 421. All were "known options within [the skilled practitioner's] technical grasp." *Id.*; *see also* Appx377 col. 14:13–16 (disclosing lookup tables); Appx440 col. 17:17–28 (disclosing arithmetic operators); Appx397 col. 9:8–13 (disclosing logical operators). Indeed, Buxton itself discussed the XOR logical operator in connection with its discrete algorithms embodiment, as discussed above.

The Board's finding of non-obviousness failed to take into account this extensive "background knowledge" as well as "the inferences and creative steps" that a skilled programmer "would employ." *KSR*, 550 U.S. at 418. Having found that Buxton's use of arithmetic operators differed from the '960 patent's use of logical operators, the Board ended its analysis—precisely the type of "blinkered focus on individual

documents" this Court has rejected. *Randall Mfg. v. Rea*, 733 F.3d 1355, 1362 (Fed. Cir. 2013). The Court should complete the analysis and hold that any "gap between" the '960 patent and Buxton "is simply not so great as to render the system nonobvious to one reasonably skilled in the art." *Dann*, 425 U.S. at 230.

III. The Board's Final Decision Rests On An Unreasoned Departure From Its Institution Decision.

The Board's final decision is arbitrary and capricious for an additional reason: it is inconsistent with the Board's decision regarding institution. The Board denied review of Martin based on the redundancy doctrine, under which the Board sometimes "declin[es] to proceed on redundant grounds in [the] absence of [a] meaningful distinction between grounds." Appx889 (citing *Liberty Mut. Ins. Co. v. Progressive Cas. Ins. Co.*, No. CBM2012-00003 (PTAB Oct. 25, 2012) (Paper 7)). On its face, Martin unquestionably describes the use of logical operators ("AND, OR, XOR, NOR, and NAND") for "image combining." Appx397 col. 9:10–13.

The Board's final decision cannot be reconciled with its decision on institution. If Gough and Buxton do not disclose logical operators, as the Board concluded, then Martin is not redundant. At a minimum, the

Board was required to provide a reasoned explanation in its final decision for the inconsistency. *See M.M. & P. Mar. Advancement, Training, Educ. & Safety Program v. Dep't of Commerce*, 729 F.2d 748, 755 (Fed. Cir. 1984). The Board offered none. That renders the Board's final written decision arbitrary and capricious. *See Motor Vehicle Mfrs. Ass'n of U.S., Inc. v. State Farm Mut. Auto. Ins. Co.*, 463 U.S. 29, 43 (1983).[3]

The institution decision stated that Google's petition had relied on Martin "without explanation as to why [it] is better than or different from the" other asserted grounds based on Gough and Buxton. Appx889. The petition demonstrated how each of the references disclosed each of the claim elements, including logical operators. *See* Appx58–59 (Gough); Appx73 (Martin); Appx90 (Buxton). And in

---

[3] To be clear, Google is challenging the Board's final written decision, not its institution decision. Because the institution decision is relevant to this Court's review of the final decision for the reasons explained above, and it is part of the record, this Court may properly consider it. *See, e.g., Ethicon Endo-Surgery, Inc. v. Covidien LP*, 812 F.3d 1023, 1029 (Fed. Cir. 2016) (review allowed where petitioner "does not challenge the institution decision, but rather alleges a defect in the final decision"); *Synopsys*, 814 F.3d at 1314 ("Synopsys does not challenge the decision to institute but rather the scope of the final decision itself. Because Synopsys challenges the final decision, we can hear this appeal.").

Google's view, no reference was "better than" the others on this point because they all disclose logical operators. Thus, it is not clear what more Google could have done in the petition. In any event, the inconsistency here—finding that Martin was redundant but then concluding that the other references did not disclose logical operators—is the Board's alone, and it renders the Board's final decision arbitrary and capricious under the cases cited above.

CONCLUSION

For the foregoing reasons, Google respectfully requests that this Court reverse the Board's decision and rule that claims 1–3, 5, 7–10, and 12–14 of the '960 patent are unpatentable. In the alternative, the Court should remand for the Board to fully consider the record and provide a reasoned explanation for its decision.

Respectfully submitted,

Dated: June 3, 2016

/s/ Daryl J. Joseffer

Daryl L. Joseffer
KING & SPALDING LLP
1700 Pennsylvania Ave., NW
Washington, DC 20006
Telephone: (202) 737-0500
Facsimile: (202) 626-3737
djoseffer@kslaw.com

Adam M. Conrad
KING & SPALDING LLP
100 N. Tryon St., Suite 3900
Charlotte, NC 28202
Telephone: (704) 503-2600
Facsimile: (704) 503-2622
aconrad@kslaw.com

CERTIFICATE OF SERVICE

I certify that on June 3, 2016, I caused the foregoing brief to be filed with the Court electronically using the CM/ECF system, which will send a notification to all counsel of record.

June 3, 2016

/s/ Daryl L. Joseffer
Daryl L. Joseffer

CERTIFICATE OF COMPLIANCE

I certify that this brief complies with the type-volume limitation of Fed. R. App. P. 32(a)(7)(B) because it contains 8,112 words, excluding the parts of the brief exempted by Fed. R. App. P. 32(a)(7)(B)(iii).

June 3, 2016

/s/ Daryl L. Joseffer
Daryl L. Joseffer